AMENDMENTS TO THE CLAIMS:

1. (Currently amended) A method of executing a linear algebra subroutine on a machine having at least one floating point unit (FPU) with one or more associated load/store units (LSU) to load data into and out of floating point registers (FRegs) of said FPU by way of a cache, said method comprising:

for an execution code controlling an operation of ~~a~~ said floating point unit (FPU) performing a linear algebra subroutine execution, inserting instructions to ~~timely~~ move data into ~~a~~ said cache providing data for said FPU so that said LSUs can move said data into said FRegs in a timely manner for said linear algebra subroutine execution, said data being prefetched into said cache from a memory in a nonstandard format predetermined to reduce a number of data streams for a level 3 processing to be three streams and to allow a multiple loading of loads into said FPU by said LSU~~, thereby improving an efficiency for said linear algebra subroutine execution~~.

2. (Previously presented) The method of claim 1, wherein said timely moving data is accomplished by scheduling move type instructions into time slots existing in a Level 3 Dense Linear Algebra Subroutine.

3. (Previously presented) The method of claim 1, wherein said linear algebra subroutine comprises a matrix multiplication operation.

4. (Currently amended) The method of claim 1, wherein said linear algebra subroutine comprises ~~a more efficient~~ an equivalent of a subroutine from LAPACK (Linear Algebra PACKage).

5. (Previously presented) The method of claim 1, wherein said linear algebra subroutine invokes a BLAS Level 3 L1 cache kernel.

6. (Currently amended) An apparatus, comprising:

a memory to store matrix data to be used for processing in a linear algebra program;

a floating point unit (FPU) to perform said processing;

a load/store unit (LSU) to load data to be processed by said FPU, said LSU loading said data into a plurality of floating point registers (FRegs); and

a cache to store data from said memory and provide said data to said FRegs,

wherein said matrix data in said memory is timely moved by inserting moving instructions to be loaded into said cache prior to a need for said data to be loaded by said LSU into said FRegs for said processing, said data being prefetched into said cache from said memory in a nonstandard format predetermined to reduce a number of data streams for a level 3 processing to be three streams and to allow a multiple loading of loads into said FPU by said LSU.

7. (Original) The apparatus of claim 6, wherein said linear algebra program comprises a matrix multiplication operation.

8. (Currently amended) The apparatus of claim 6, wherein said linear algebra program comprises ~~a more efficient~~ an equivalent of a subroutine from a LAPACK (Linear Algebra PACKage).

9. (Previously presented) The apparatus of claim 6, wherein said processing comprises invoking a BLAS Level 3 L1 cache kernel.

10. (Canceled)

11. (Currently amended) The apparatus of claim ~~10~~ 6, wherein said moving instructions are inserted into time slots existing in a Level 3 Dense Linear Algebra Subroutine.

12. (Currently amended) A computer-readable storage medium tangibly embodying a program of machine-readable instructions executable by a digital processing apparatus to perform a method of executing linear algebra subroutines on a machine having at least one floating point unit (FPU) with one or more associated load/store units (LSUs) to load data into and out of floating point registers (FRegs) of said FPU by way of a cache, said method comprising:

for an execution code controlling an operation of a floating point unit (FPU) performing a linear algebra subroutine execution, inserting instructions to timely move data into ~~a~~ said cache providing said data into said FPU,

wherein said data is prefetched into said cache from a memory in a nonstandard format predetermined to reduce a number of data streams for a level 3 processing to be three streams and to allow a multiple loading of loads into said FPU by said LSUs ~~thereby improving an efficiency for said linear algebra subroutine execution~~.

13. (Previously presented) The computer-readable storage medium of claim 12, wherein said timely moving data is accomplished by inserting move type instructions into time slots existing in a Level 3 Dense Linear Algebra Subroutine.

14. (Previously presented) The computer-readable storage medium of claim 12, wherein said linear algebra subroutine comprises a matrix multiplication operation.

15. (Currently amended) The computer-readable storage medium of claim 12, wherein said linear algebra subroutine comprises ~~a more efficient~~ an equivalent of a subroutine from LAPACK (Linear Algebra PACKage).

16. (Previously presented) The computer-readable storage medium of claim 12, wherein said linear algebra subroutine invokes a BLAS Level 3 L1 cache kernel.

17. (Currently amended) A method of providing a service involving at least one of solving and applying a scientific/engineering problem, said method comprising at least one of:

using a linear algebra software package that computes one or more matrix subroutines, wherein said linear algebra software package generates an execution code controlling an operation of a floating point unit (FPU) performing a linear algebra subroutine execution, such that instructions are inserted to timely move data into a cache providing data for said FPU, said data being prefetched from a memory in a nonstandard format predetermined to reduce a number of data streams for a level 3 processing to be three streams and to permit a multiple loading of loads into said FPU ~~thereby improving an efficiency for said linear algebra subroutine execution~~;

providing a consultation for solving a scientific/engineering problem using said linear algebra software package;

transmitting a result of said linear algebra software package on at least one of a network, a signal-bearing medium containing machine-readable data representing said result, and a printed version representing said result; and

receiving a result of said linear algebra software package on at least one of a network, a signal-bearing medium containing machine-readable data representing said result, and a printed version representing said result.

18. (Currently amended) The method of claim 17, wherein said linear algebra subroutine comprises ~~a more efficient~~ an equivalent of a subroutine from LAPACK (Linear Algebra PACKage).

19. (Previously presented) The method of claim 17, wherein said linear algebra subroutine invokes a BLAS Level 3 L1 cache kernel.

20. (Canceled)